UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/631,205	07/31/2003	Gerard Chauvel	TI-35432	3321

23494 7590 09/28/2007
TEXAS INSTRUMENTS INCORPORATED
P O BOX 655474, M/S 3999
DALLAS, TX 75265

EXAMINER	
RUTZ, JARED IAN	
ART UNIT	**PAPER NUMBER**
2187	

NOTIFICATION DATE	DELIVERY MODE
09/28/2007	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

uspto@ti.com
uspto@dlemail.itg.ti.com

Office Action Summary	**Application No.** 10/631,205	**Applicant(s)** CHAUVEL ET AL.	
	Examiner Jared I. Rutz	**Art Unit** 2187	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *13 July 2007*.

2a)□ This action is **FINAL**. 2b)☒ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-14 and 16-20* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-14 and 16-20* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application

6) □ Other: _____.

DETAILED ACTION

1. Claims 1-14 and 16-20, as amended on 7/13/2007 with the filing of a Request for Continued Examination, are pending in the instant application. Applicant's arguments submitted 7/13/2007 have been carefully and fully considered, but are only found partially persuasive.

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. **Claims 1-3, 6, 8-11, 14, 16-18, and 20** are rejected under 35 U.S.C. 103(a) as being unpatentable over Adams, III et al. (US 6,151,661) in view of Lopriore (Line fetch/prefetch in a stack cache memory).

4. **Claim 1** is taught by Adams as:

a. *A method of managing memory, comprising: issuing a data request to remove data.* Column 4 lines 55-56 show a POP operation issued by processor 12.

b. *And if the data being removed corresponds to a predetermined word in the dirty cache line, queuing the dirty cache line for replacement.* Adams at column 4 line 66 through column 5 line 6 shows that if the data word to be popped is the highest address word in the cache line, the other words in the cache line are necessarily invalid, and the processor is instructed to invalidate the cache line. When the cache line is invalidated, it is left empty and made available for other valid data.

5. Adams teaches that if the data word to be POPPED is the highest address word in the cache line, that word is the last possible word to be POPPED in that particular cache line and the entire cache line is invalidated (column 5 lines 2-4). Adams also states that the cache line is made available for other valid data (column 5 lines 4-6) Adams makes no exceptions for dirty/not dirty lines. However, Adams does not explicitly teach that a dirty cache line is invalidated, or that the invalidated cache line is not written to memory after POPPING the highest address word of a cache line.

6. With respect to claim 1, Lopriore teaches:

c. *And not writing the dirty cache line to a memory external to a processor.* In the third paragraph of the second column of page 547, which continues on page 548, Lopriore states "*when the stack shrinks, the cache lines reserved for the discarded stack portions at the stack top become free. The cache contains no valid information for these lines, so we do not have to copy the contents of these lines to the primary memory*"

7. Adams and Lopriore are analogous art because they are from the same field of endeavor, computer data cache design.

8. At the time of the invention it would have been obvious to a person of ordinary skill in the art to not write a dirty cache line to main memory when the highest address

word in the cache line is POPPED, as the cache line does not contain any valid data (Adams column 5, lines 2-4; Lopriore page 547, second column, line 39 through page 548, first column, line 2).

9. The motivation for doing so would have been to save unnecessary writes to main memory (Lopriore page 547, second column, lines 1-14; also a goal of Adams, column 1 line 54 through column 2 line 11).

10. Therefore, it would have been obvious to combine Lopriore with Adams for the benefit of eliminating unnecessary memory cycles to obtain the invention as specified in **claims 1-3, 6, and 8**.

11. **Claim 2** is taught by Adams as:

d. *The method of claim 1, wherein the predetermined word is the first word in the dirty cache line.* Column 4 line 66 to column 5 line 6 shows that if the popped data word is the highest address word in the cache line, the processor invalidates the cache line. As shown in figure 2, in the direction of a push operation the highest address word is first in the cache line.

12. **Claim 3** is taught by Adams as:

e. *The method of claim 2, wherein the dirty cache line is invalidated.* Column 4 line 66 to column 5 line 6 shows that if the popped data word is the highest address word in the cache line, the processor invalidates the cache line.

13. **Claim 6** is taught by Adams as:

f. *The method of claim 1, wherein the predetermined word is the last word in the dirty cache line.* Column 4 line 66 to column 5 line 6 shows that if the popped data word is the highest address word in the cache line, the processor invalidates the cache line. As shown in figure 2, in the direction of a pop operation the highest address word is the last word in the cache line.

14. **Claim 8** is taught by Adams as:

g. *The method of claim 1, further comprising invalidating the dirty cache line if the predetermined word in the dirty cache line is the first word.* Column 4 line 66 to column 5 line 6 shows that if the popped data word is the highest address word in the cache line, the processor invalidates the cache line. As shown in figure 2, in the direction of a push operation the highest address word is first in the cache line.

15. **Claim 9** is taught by Adams as:

h. *A system, comprising: a memory.* Figure 1 items 14 and 18.

i. *A controller coupled to the memory.* Figure 1 item 16.

j. *And a stack that exists in the memory.* Column 3 lines 54-57 show that stacks are stored within the cache.

k. *Wherein the memory further comprises a cache memory and a main memory.* Figure 1 item 14 is a cache and item 18 is a main memory.

l. *Wherein, if data being removed comprises a predetermined word in a dirty cache line, the controller queues the dirty cache line to be overwritten.* Adams at column 4 line 66 through column 5 line 6 shows that if the data word to be popped is the highest address word in the cache line, the other words in the cache line are necessarily invalid, and the processor is instructed to invalidate the cache line. When the cache line is invalidated, it is left empty and made available for other valid data.

16. Adams teaches that if the data word to be POPPED is the highest address word in the cache line, that word is the last possible word to be POPPED in that particular cache line and the entire cache line is invalidated (column 5 lines 2-4). Adams also states that the cache line is made available for other valid data (column 5 lines 4-6) Adams makes no exceptions for dirty/not dirty lines. However, Adams does not explicitly teach that a dirty cache line is invalidated, or that the invalidated cache line is not written to memory.

17. With respect to claim 9, Lopriore teaches:

m. *And wherein the dirty cache line is not saved to the main memory.* In the third paragraph of the second column of page 547, which continues on page 548, Lopriore states "*when the stack shrinks, the cache lines reserved for the discarded stack portions at the stack top become free. The cache contains no valid information for these lines, so we do not have to copy the contents of these lines to the primary memory*"

18. Adams and Lopriore are analogous art because they are from the same field of endeavor, computer data cache design.

19. At the time of the invention it would have been obvious to a person of ordinary skill in the art to not write a dirty cache line to main memory when the highest address word in the cache line is POPPED, as the cache line does not contain any valid data (Adams column 5, lines 2-4; Lopriore page 547, second column, line 39 through page 548, first column, line 2).

20. The motivation for doing so would have been to save unnecessary writes to main memory (Lopriore page 547, second column, lines 1-14; also a goal of Adams, column 1 line 54 through column 2 line 11).

21. Therefore, it would have been obvious to combine Lopriore with Adams for the benefit of eliminating unnecessary memory cycles to obtain the invention as specified in **claims 9-11, 14 and 16**.

22. **Claim 10** is taught by Adams as:

n. *The system of claim 9, wherein the predetermined word is the first word in the dirty cache line*. Column 4 line 66 to column 5 line 6 shows that if the popped data word is the highest address word in the cache line, the processor invalidates the cache line. As shown in figure 2, in the direction of a push operation the highest address word is first in the cache line.

23. **Claim 11** is taught by Adams as:

o. *The system of claim 10, wherein the controller queues the dirty cache line to be overwritten by invalidating the dirty cache line.* Column 5 lines 2-4 show that the cache line is invalidated.

24. **Claim 14** is taught by Adams as:

p. *The system of claim 9, wherein the predetermined word is the last word in the dirty cache line.* Column 4 line 66 to column 5 line 6 shows that if the popped data word is the highest address word in the cache line, the processor invalidates the cache line. As shown in figure 2, in the direction of a pop operation the highest address word is the last word in the cache line.

25. **Claim 16** is taught by Adams as:

q. *The system of claim 9, wherein the dirty cache line is invalidated if the predetermined word in the dirty cache line is the first word.* Column 4 line 66 to column 5 line 6 shows that if the popped data word is the highest address word in the cache line, the processor invalidates the cache line. As shown in figure 2, in the direction of a push operation the highest address word is first in the cache line.

26. **Claim 17** is taught by Adams as:

r. *A system, comprising: a processor that executes stack-based instructions.* Column 1 lines 55-58.

s. *A cache controller coupled to the processor.* Stack cache support logic 16, see column 4 lines 38-42.

t. *And a cache memory coupled to and controlled by said cache controller.* Figure 1 item 14.

u. *Said cache memory storing at least a portion of a stack.* Column 3 lines 54- 57 shows that stacks are stored within the cache.

v. *Said stack having a top and a read access of the stack causes the top of the stack to be read.* Column 1 lines 26-32.

w. *Wherein, if the processor reads a value from the top of the stack that comprises a word at a predetermined location within a dirty cache line in said cache memory, the cache controller queues said dirty cache line for replacement.* Adams at column 4 line 66 through column 5 line 6 shows that if the data word to be popped is the highest address word in the cache line, the other words in the cache line are necessarily invalid, and the processor is instructed to invalidate the cache line. When the cache line is invalidated, it is left empty and made available for other valid data.

27. Adams teaches that if the data word to be POPPED is the highest address word in the cache line, that word is the last possible word to be POPPED in that particular cache line and the entire cache line is invalidated (column 5 lines 2-4). Adams also states that the cache line is made available for other valid data (column 5 lines 4-6). Adams makes no exceptions for dirty/not dirty lines. However, Adams does not

explicitly teach that a dirty cache line is invalidated, or that the invalidated cache line is not written to memory.

28. With respect to claim 17, Lopriore teaches:

x. *And the dirty cache line is not written to a memory external to the processor.* In the third paragraph of the second column of page 547, which continues on page 548, Lopriore states "*when the stack shrinks, the cache lines reserved for the discarded stack portions at the stack top become free. The cache contains no valid information for these lines, so we do not have to copy the contents of these lines to the primary memory*"

29. Adams and Lopriore are analogous art because they are from the same field of endeavor, computer data cache design.

30. At the time of the invention it would have been obvious to a person of ordinary skill in the art to not write a dirty cache line to main memory when the highest address word in the cache line is POPPED, as the cache line does not contain any valid data (Adams column 5, lines 2-4; Lopriore page 547, second column, line 39 through page 548, first column, line 2).

31. The motivation for doing so would have been to save unnecessary writes to main memory (Lopriore page 547, second column, lines 1-14; also a goal of Adams, column 1 line 54 through column 2 line 11).

32. Therefore, it would have been obvious to combine Lopriore with Adams for the benefit of eliminating unnecessary memory cycles to obtain the invention as specified in **claims 17, 18, and 20**.

33. **Claim 18** is taught by Adams as:

y. *The system of claim 17 wherein said predetermined location is selected from a group consisting of the first word and the last word of the line.* Column 4 line 66 to column 5 line 6 shows that if the popped data word is the highest address word in the cache line, the processor invalidates the cache line. As shown in figure 2, in the direction of a pop operation the highest address word is the last word in the cache line.

34. **Claim 20** is taught by Adams as:

z. *The system of claim 17, wherein the cache controller queues the dirty cache line for replacement by invalidating the dirty cache line.* Column 5 lines 2-4 show that the cache line is invalidated.

35. **Claims 4-5, 7, 12, 13, and 19** are rejected under 35 U.S.C. 103(a) as being unpatentable over Adams, III et al. (cited supra) in view of Lopriore (cited supra) and further in view of Handy (The Cache Memory Book).

36. **Claim 4** is taught by Adams and Lopriore as shown supra with respect to claim 2.

37. With respect to claim 4, Handy teaches:

aa. *The method of claim 2, wherein the dirty cache line is queued for replacement by a replacement policy when a read hit occurs on the first word of* *the dirty cache line.* With respect to claim 4 Handy teaches that if the cache is a multi-way cache, an algorithm must be used to select a line to be replaced, page 57 lines 3-14.

38. Adams, Lopriore, and Handy are analogous art because they are from the same field of endeavor, computer data cache design.

39. At the time of the invention it would have been obvious to use a multi-way set associative cache in the system of Adams.

40. The motivation for doing so would have been to lessen thrashing while still improving the hit rate over a direct mapped cache, Handy page 54 lines 1-18.

41. Therefore it would have been obvious to combine the multi-way set associative cache of Handy with Adams for the benefit of reducing thrashing while improving hit rate to obtain the invention as specified in **claims 4-5**.

42. **Claim 5** is taught by Handy as:

bb. *The method of claim 4, wherein the replacement policy is a least recently used (LRU) policy.* Page 57 lines 10-14 teaches the use of a LRU replacement algorithm.

43. **Claim 7** is taught by Adams and Lopriore as shown supra with respect to claim 1.

44. With respect to claim 7 Handy teaches:

cc. *Wherein queuing the dirty cache line for replacement comprises designating the dirty cache line as least-recently-used (LRU).* Handy page 57 lines 9-10 state "*Ideally, any stale piece of cached data which is no longer needed by the processor would be chosen to be overwritten.*" As the cache line is invalidated when the highest address word in the cache line is popped, that cache line no longer contains data which is needed by the processor. Page 57 lines 10-14 teaches the use of a LRU replacement algorithm.

45. Adams, Lopriore, and Handy are analogous art because they are from the same field of endeavor, computer data cache design.

46. At the time of the invention, it would have been obvious to one of ordinary skill in the art to designate the invalidated dirty cache line as least-recently-used.

47. The motivation for doing so would have been to prevent evicting useful data, which may still be useful to the processor, from the cache.

48. Therefore, it would have been obvious to one of ordinary skill in the art to designate the invalidated cache line as least recently used, for the benefit of avoiding the eviction of useful data, to obtain the invention as specified in **claim 7**.

49. **Claim 12** is taught by the combination of Adams and Lopriore as shown supra with respect to claim 11.

50. Although the combination of Adams and Lopriore teaches that the dirty cache line is made available for other valid data (Adams column 5 lines 4-6), the combination of Adams and Lopriore does not explicitly teach queuing the invalidated cache line for replacement by a least-recently-used replacement policy.

51. With respect to claim 12, Handy teaches:

dd. *The system of claim 11, wherein the invalidated cache line is queued to be overwritten by a least-recently-used (LRU) replacement policy.* At page 57 lines 3-14, Handy teaches that if the cache is a multi-way cache, an algorithm must be used to select a line to be replaced. Page 57 lines 10-14 teaches the use of a LRU replacement algorithm.

52. The combination of Adams and Lopriore and Handy are analogous art because they are from the same field of endeavor, computer data cache design.

53. At the time of the invention it would have been obvious to use a multi-way set associative cache with a least-recently-used replacement policy in the system of Adams and Lopriore.

54. The motivation for using a multi-way set associative cache doing so would have been to lessen thrashing while still improving the hit rate over a direct mapped cache, Handy page 54 lines 1-18. The motivation for using a LRU replacement policy would have been that a processor is much more likely to need to access a memory location which it accessed ten cycles ago than one which it accessed ten thousand cycles ago, Handy page 7 lines 18-22. By replacing the least recently used cache line, a LRU replacement policy reduces the likelihood of evicting data that the processor will need.

55. Therefore it would have been obvious to combine Handy with the combination of Adams and Lopriore for the benefit of reducing thrashing while improving hit rate to obtain the invention as specified in **claim 12**.

56. **Claims 13 and 19** are taught by the combination of Adams and Lopriore as shown supra with respect to claims 9 and 17, respectively.

57. Although the combination of Adams and Lopriore teaches that the dirty cache line is made available for other valid data (Adams column 5 lines 4-6), the combination of Adams and Lopriore does not explicitly teach that the controller designates the dirty cache line as the least recently used cache line.

58. With respect to claims 13 and 19, Handy teaches:

ee. *Wherein the controller queues the dirty cache line to be overwritten by designating the dirty cache line as the least-recently-used (LRU) cache line.* Page 57 lines 9-10 state "*Ideally, any stale piece of cached data which is no longer needed by the processor would be chosen to be overwritten.*" As the cache line is invalidated when the highest address word in the cache line is popped, that cache line no longer contains data which is needed by the processor.

59. The combination of Adams and Lopriore and Handy are analogous art because they are from the same field of endeavor, computer data cache design.

60. At the time of the invention, it would have been obvious to one of ordinary skill in the art to designate the invalidated dirty cache line for replacement.

61. The motivation for doing so would have been to prevent evicting useful data, which may still be useful to the processor, from the cache.

62. Therefore, it would have been obvious to combine Handy with the combination of Adams and Lopriore for the benefit of avoiding evicting useful data to obtain the invention as specified in **claims 13 and 19**.

Response to Arguments

63. Applicant's arguments submitted 7/13/2007 have been carefully and fully considered, but are only found partially persuasive.

64. **First point of Argument**

65. In the second paragraph beginning on page 6 of the arguments submitted 7/13/2007, applicant argues with respect to the rejection of claims 1-8 under 35 USC 112 first paragraph that the limitation "determining whether the data is being removed from a dirty cache line in a cache memory" have been removed from claim 1. This is sufficient to overcome the rejection of claims 1-8 under 35 USC 112 first paragraph. Accordingly, said rejection is withdrawn.

66. **Second point of Argument**

67. In the third paragraph beginning on page 6 of the arguments submitted 7/13/2007, applicant argues with respect to the rejection of claims 1-8, 12-13, and 19 under 35 USC 103(a):

> ff. *"Amended claim 1 requires 'if the data being removed corresponds to a predetermined word in a dirty cache line in a cache memory, queuing the dirty cache line for replacement…' (emphasis added). The combination of Adams,*

> *Lopriore and Handy fails to teach or suggest this limitation. In paragraphs 15 and 1 of the Final Office Action, the Examiner admits that Adams fails to disclose the above limitation in the context of dirty cache lines (see, in particular, the last sentence of paragraph 1). The Examiner also fails to point out where Lopriore and Handy teach or even suggest the above limitations in the context of dirty cache lines. It appears that Lopriore and Handy fail to teach or even suggest this limitation in the context of dirty cache lines. Claim 1 and dependent claims 2-8 are patentable over the combination of Adams, Lopriore and Handy for at least this reason."*

68. The Examiner respectfully disagrees. Adams column 4 line 66 through column 5 line 6 states "*If, however, the data word to be POPPED is the highest address word in the cache line, that word is the last possible word to be POPPED in that particular cache line and the other words in the cache line are necessarily invalid. The highest address word is passed to the processor and the processor is instructed to invalidate the cache line. In this manner, the cache line is left empty and made available for other valid data.*" The Examiner submits that this provides an explicit teaching of queuing a cache line for replacement if the data being removed is a predetermined word in a cache line. The Examiner notes that this section does not contain an explicit teaching of invalidating a dirty cache line. However, it is maintained that this section inherently applies to a dirty cache line, because it is irrelevant if a cache line is dirty or not dirty, it is necessarily invalid, i.e. contains no valid data, and therefore can be queued for replacement.

69. **Third point of Argument**

70. In the first paragraph beginning on page 7 of the arguments submitted 7/13/2007, applicant argues with respect to the rejection of claims 1-8, 12-13, and 19 under 35 USC 103(a):

> gg. "*Claim 1 is patentable for an additional reason. In particular, claim 1 requires '...and not writing the dirty cache line to a memory external to a processor.' The combination of Adams, Lopriore and Handy fails to teach or suggest this limitation. In paragraphs 17-19 of the Final Office Action, the Examiner asserts that it would have been obvious to not write a dirty cache line to main memory when the highest address word in the cache line is POPPED, as the cache line does not contain any valid data, and cites Adams and Lopriore for support. However, Applicants remind the Examiner that the cited teachings of the references are not directed to dirty cache lines, as is claim 1. Dirty cache lines generally are designated as 'dirty' because they contain data which needs to be written to memory before the dirty cache line is invalidated or otherwise cleared for other data. Claim 1 distinguishes over prior art at least because it requires that the dirty cache line data not be written to memory before the dirty cache line is queued for replacement. In the apparent absence of a dirty cache line context in the cited references, the Examiner cannot presume sua sponte that the references' teachings would hold true for dirty cache lines.*"

71. The Examiner respectfully disagrees. Adams, at column 2 lines 47-51 states "*If, however, the data word to be POPPED is the highest address word in the cache line,*

that word is the last possible word to be popped in that particular cache line and the other words in the cache line are ***necessarily*** *invalid.*" As such, it does not matter if the cache line is dirty or not dirty, it is necessarily invalid. Lopriore, at line 39 of the second column of page 547 through line 2 of the first column of page 548, states "*When the stack shrinks, the cache lines reserved for the discarded stack portions at the stack top become free. The cache contains no valid information for these lines, so we do not have to copy the contents of these lines to the primary memory*". The Examiner respectfully submits that Applicant is not considering the teachings of Adams and Lopriore as a whole. As is clear from Adams, when the highest address word of a cache line is popped, the other words in that line do not contain valid data. As is known by one of ordinary skill in the art, and taught by Adams at column 1 lines 26-32, a stack is a last-in first-out data structure. This means that only the item at the top of the stack can be accessed through a pop operation, and an item pushed onto the stack becomes the new top of the stack. Once an item is popped from the stack, it is logically removed from the stack. As shown in figure 2 of Adams, items at lower address words are "higher" in the stack than the highest addressed word. When the item stored at the highest address word is popped, the items stored in the lower addressed words must have already been popped, or never contained stack data, as the highest address word would not be on top of the stack to be popped if there were items above it in the stack. Therefore, as explicitly stated by Adams at column 2 lines 47-51, when the highest address word is popped, the cache line no longer contains valid data.

72. As is known by one of ordinary skill in the art, and taught by Lopriore at lines 2-8 of the second column of page 547, a cache line is marked "dirty" when it has been written to, as its data no longer matches the corresponding portion of primary memory. Typically, dirty cache lines need to be written back to primary memory, as the contents of the data stored in the cache differ from the data stored in the corresponding location in memory, and if the data were not written back to primary memory changed data would be lost when the cache line was overwritten.

73. However, as taught by Adams at column 2 lines 47-51, when the highest address word is popped, the cache line no longer contains valid data. As taught by Lopriore at line 39 of the second column of page 547 through line 2 of the first column of page 548, when a cache line contains no valid data, it is not necessary to copy the contents of the line to primary memory. Accordingly, the Examiner submits that the combination of Adams and Lopriore teach the limitation "*and not writing the dirty cache line to a memory external to a processor.*"

Conclusion

74. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

75. Additional pages from the Handy reference (The Cache Memory Book) are submitted to further show that it is known to one of ordinary skill in the art that cache lines that are not valid do not contain valid information. Figure 2.14 identifies the data stored in lines with a valid bit set to 0 as garbage and figure 4.8 shows that when the

line is invalid, the line is invalid whether the dirty bit indicates dirty or not dirty (indicated by an X, known to one of ordinary skill in the art to indicate a "don't care" condition).

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jared I. Rutz whose telephone number is (571) 272-5535. The examiner can normally be reached on M-F 8:00 AM - 4:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Donald Sparks can be reached on (571) 272-4201. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Jared I Rutz
Examiner
Art Unit 2187

Brian R. Peugh
Primary Examiner
9/20/07

jir